

CODE FOR KERNEL GRAPH CUT SEGMENTATION

This code implements multi-region graph cut image segmentation according to the kernel-mapping formulation in M. Ben Salah, A. Mitiche, and I. Ben Ayed, *Multiregion Image Segmentation by Parametric Kernel Graph Cuts*, IEEE Transactions on Image Processing, 20(2): 545-557 (2011).

The code uses Veksler, Boykov, Zabih and Kolmogorov's implementation of the Graph Cut algorithm. Written in C++, the graph cut algorithm comes bundled with a MATLAB wrapper by Shai Bagon (Weizmann), which has to be downloaded from the following link (Matlab Wrapper for Graph Cuts):

http://www.wisdom.weizmann.ac.il/~bagon/matlab_code/GCmex1.9.tar.gz

The kernel-mapping part was implemented in MATLAB by M. Ben Salah (University of Alberta). If you use this code, please cite the papers mentioned in the accompanying bib file (citations.bib).

The kernel-mapping formulation can handle various type of images, including color photographs as well as data corrupted by a strong multiplicative noise as in remote sensing synthetic aperture radar (SAR) or medical imaging ultrasound. It is an efficient and flexible alternative to explicit modeling of imaging noise using standard distributions (e.g., Gamma, Raleigh, Exponential, Gaussian, Weibull, etc.).

Requirements:

This code was tested with:

- MATLAB Version: 7.12.0.635 (R2011a) for 32-bit wrapper
- Microsoft Visual C++ 2010 Express,

and requires the following C++/MATLAB files (included in GCmex1.9.tar.gz), which can be downloaded from the above Matlab-Wrapper link:

- block.h
- energy.h
- GCOptimization.cpp

- GCOptimization.h
- graph.cpp
- graph.h
- GraphCut.cpp
- GraphCut.h
- GraphCut.m
- GraphCutConstr.cpp
- GraphCutMex.cpp
- LinkedList.cpp
- LinkedList.h
- maxflow.cpp

Generating the mex files in MATLAB (all C++/MATLAB files have to be under the same folder):

- `mex -g GraphCutConstr.cpp graph.cpp GCOptimization.cpp GraphCut.cpp LinkedList.cpp maxflow.cpp`
- `mex -g GraphCutMex.cpp graph.cpp GCOptimization.cpp GraphCut.cpp LinkedList.cpp maxflow.cpp`

Usage:

Script *Kernel_GraphCuts_Examples.m* shows examples of how to use the implementation, including a color image, a SAR image corrupted with a strong multiplicative speckle noise, and a brain image. It contains the following main inputs and parameters (Note: The RBF-kernel parameters are given in function *kernel_RBF.m*):

- *path*: The path of the image to be segmented
- *alpha*: The weight of the smoothness constraint
- *k*: The number of regions